



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/847,781	05/01/2001	Martin Matula	SUN-P5775	6697

7590 01/19/2005

David B. Ritchie  
Thelen Reid & Priest LLP  
P. O. Box 640640  
San Jose, CA 95164-0640

EXAMINER

CAO, DIEM K

ART UNIT PAPER NUMBER

2126

DATE MAILED: 01/19/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application No.

09/847,781

Applicant(s)

MATULA ET AL.

Examiner

Diem Cao

Art Unit

2127

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 23 August 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-66 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-66 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_

## DETAILED ACTION

### *Double Patenting*

1. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and, *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

2. Claims 1, 17, 34, and 50 are provisionally rejected under the judicially created doctrine of double patenting over claims 67, 83, 100 and 111 of copending Application No. 09/848,392. This is a provisional double patenting rejection since the conflicting claims have not yet been patented.

The subject matter claimed in the instant application is fully disclosed in the referenced copending application and would be covered by any patent granted on that copending application since the referenced copending application and the instant application are claiming common subject matter, as follows: Both the current application and the copending application disclose steps/means for receiving and implementing a JMI implementation from a request associated with a metamodel. The cited claims would not be restricted from one another because both performing similar steps/actions for generating a dynamic implementation of a JMI interface.

Art Unit: 2127

Furthermore, there is no apparent reason why applicant would be prevented from presenting claims corresponding to those of the instant application in the other copending application. See *In re Schneller*, 397 F.2d 350, 158 USPQ 210 (CCPA 1968). See also MPEP § 804.

It appears that Applicant justifies the distinction of the two applications because one application deals with a method and another deals with an apparatus. However, in order for the claims to be distinct and therefore restricted from one another, the method must be able to be practiced by a materially different apparatus or the apparatus must implement a materially different method. Neither is the case here. The apparatus claims are claims having means / apparatus elements that denote the exact same steps and functionality as disclosed in the method. Therefore, the apparatus cannot perform another materially different method. Similarly, since the method is not limited to any particular system/framework, it can operate in any materially apparatus as long as the cited method is performed, which is exactly what is denoted in the apparatus claimed in the co-pending application. The examiner is willing to wait until the claims are in condition for allowance for filing the terminal disclaimer, however, the rejection must be maintained in any action sent until this matter is resolved.

### ***Claim Rejections - 35 USC § 103***

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the

Art Unit: 2127

invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claims 1, 2, 5, 9, 17-19, 22, 26, 34, 35, 38, 42, 50-52, 55 and 59 are rejected under 35 U.S.C. 103(a) as being unpatentable over Applicant's Admitted Prior Art (APA) in view of HELLGREN (U.S. Patent 6,023,579).

As to claim 1, APA teaches a method for dynamic implementation of a Java Metadata Interface, JMI, to a metamodel, the method comprising: receiving a JMI implementation request (message with a metamodel received by repository), the request associated with a metamodel (contains metamodel), the metamodel comprising at least one package (which is used to generate a package proxy interface), the package comprising at least one class (which is used to generate a class proxy interface), the class comprising at least one attribute, reference, or operation (get or set methods for each attribute and reference); implementing a package proxy JMI interface when the request comprises a package proxy request; implementing a class proxy JMI interface when the request comprises a class proxy request; and implementing a class instance JMI interface when the request comprises a class instance request (via automatically generating Java metadata interfaces when a repository receives a metamodel such that a package proxy interface is generated for each object of type "Package", a class proxy interface is generated for each object of type "class", and an instance interface is generated for each object of type "class") (pg. 6, line 20 – pg. 7, line 18). However, APA does not teach that the interfaces are dynamically implemented.

HELLGREN teaches that interfaces to a meta-model are dynamically generated based on the classes by a generator (col. 3, lines 16-25; col. 4, lines 47-53) such that the client is in any particular type of client programming language (col. 3, lines 1-3) and the system determines whether the class has been generated or not in order to build the interface (col. 5, lines 11-25). It would be obvious that the interfaces are Java interfaces since the client is implemented in any programming language, the client accesses the metadata through the interface and Java is a well known programming language to one skilled in the art. In addition, it would be obvious in view of the combination that all types of APA interfaces, i.e. package proxy interface, class proxy interface, and a class instance interface is dynamically implemented by the generator in order to access the meta-model as disclosed in HELLGREN. Therefore, it would be obvious to combine the teachings of APA with the teachings of HELLGREN in order to facilitate the automatic generation of interface definition language and server implementation code for distributed access to objects in a repository (col. 2, lines 56-61).

As to claims 2, 5, and 9, APA teaches implementing a package proxy interface, class proxy JMI interface or class instance JMI interface comprises: generating and creating new instructions/instances for a class that implements the particular interface (via the factory methods); and returning the instance (via the repository user compiling the coded JMI interface implementations and using the compiled JMI interface implementations to access the metamodel). However, APA does not teach that the instructions are bytecodes. Official Notice is taken in that it is well known in the art that

Java instructions are bytecodes and therefore it would be obvious to one skilled in the art at the time of the invention that in order to generate Java interfaces, one generates the bytecodes for the interfaces (pg. 6, line 20 – pg. 7, line 18).

As to claim 17, APA teaches a method for dynamic implementation of a Java Metadata Interface, JMI, comprising: receiving a JMI implementation request (message with a metamodel received by repository), the request associated with a metamodel (contains metamodel), the metamodel comprising at least one package (which is used to generate a package proxy interface), the package comprising at least one class (which is used to generate a class proxy interface), the class comprising at least one attribute, reference or operation (get or set methods for each attribute or reference); implementing a JMI interface when the JMI interface is unimplemented (implement JMI interface) (“a repository user manually develops the software implementation for the JMI interfaces generated...the repository user compiles the coded JMI interface implementations.”); and executing a stored JMI interface implementation when the JMI interface is implemented (“the repository user uses the compiled JMI implementations to access the metamodel”) (pg. 6, line 20 – pg. 7, line 18). However, APA does not teach that the interfaces are dynamically implemented.

HELLGREN teaches that interfaces to a meta-model are dynamically generated based on the classes by a generator (col. 3, lines 16-25; col. 4, lines 47-53) such that the client is in any particular type of client programming language (col. 3, lines 1-3) and the system determines whether the class has been generated or not in order to build the

Art Unit: 2127

interface (col. 5, lines 11-25). It would be obvious that the interfaces are Java interfaces since the client is implemented in any programming language, the client accesses the metadata through the interface and Java is a well known programming language to one skilled in the art. In addition, it would be obvious in view of the combination that all types of APA interfaces, i.e. package proxy interface, class proxy interface, and a class instance interface is dynamically implemented by the generator in order to access the meta-model as disclosed in HELLGREN. Therefore, it would be obvious to combine the teachings of APA with the teachings of HELLGREN in order to facilitate the automatic generation of interface definition language and server implementation code for distributed access to objects in a repository (col. 2, lines 56-61).

As to claim 18, APA teaches implementing a JMI interface when the JMI interface is unimplemented (implement JMI interface) ("a repository user manually develops the software implementation for the JMI interfaces generated...the repository user compiles the coded JMI interface implementations."); and executing a stored JMI interface implementation when the JMI interface is implemented ("the repository user uses the compiled JMI implementations to access the metamodel") (pg. 6, line 20 – pg. 7, line 18). APA also teaches that a package proxy interface, class proxy interface and instance interface are generated, compiled and used to access a metamodel (pg. 6, line 20 – pg. 7, line 18). It is inherent in the APA that the interfaces are implemented and



executed, i.e. used, to access the metamodel. However, APA does not teach that the interfaces are dynamically implemented.

HELLGREN teaches that interfaces to a meta-model are dynamically generated based on the classes by a generator (col. 3, lines 16-25; col. 4, lines 47-53) such that the client is in any particular type of client programming language (col. 3, lines 1-3) and the system determines whether the class has been generated or not in order to build the interface (col. 5, lines 11-25). It would be obvious that the interfaces are Java interfaces since the client is implemented in any programming language, the client accesses the metadata through the interface and Java is a well known programming language to one skilled in the art. In addition, it would be obvious in view of the combination that all types of APA interfaces, i.e. package proxy interface, class proxy interface, and a class instance interface is dynamically implemented by the generator in order to access the meta-model as disclosed in HELLGREN.

As to claims 19, 22 and 26, APA teaches implementing a package proxy interface, class proxy JMI interface or class instance JMI interface comprises: generating and creating new instructions/instances for a class that implements the particular interface (via the factory methods); and returning the instance (via the repository user compiling the coded JMI interface implementations and using the compiled JMI interface implementations to access the metamodel). However, APA does not teach that the instructions are bytecodes. Official Notice is taken in that it is well known in the art that Java instructions are bytecodes and therefore it would be

Art Unit: 2127

obvious to one skilled in the art at the time of the invention that in order to generate Java interfaces, one generates the bytecodes for the interfaces (pg. 6, line 20 – pg. 7, line 18).

As to claims 34, 35, 38, and 42, reference is made to a program readable device that corresponds to the method of claims 1, 2, 5 and 9 and is therefore met by the rejection of claims 1, 2, 5 and 9 above.

As to claims 50-52, 55 and 59, reference is made to a program readable device that corresponds to the method of claims 17-19, 22 and 26 and is therefore met by the rejection of claims 17-19, 22 and 26 above.

***Important Observed Distinctions made by the Examiner***

The examiner notes that Applicant's invention intends for the JMI interfaces to be **dynamically implemented automatically at run-time** (pg. 15, lines 22-24) **wherein the implementation are performed separately and independently from one another based on a type of request received** (via various different implementers by different requests received) (see also pg. 27, lines 19-21). However, the claims, in particular claims 1 and 34, do not illustrate this distinction. The preamble detailing the dynamic implementation of a JMI also does not teach this since the cited steps of the method only state a step of implementing. Performance of an implementing step in any manner to one skilled in the art therefore achieves dynamic implementation of a JMI. If

Art Unit: 2127

Applicant agrees with the Examiner's assessment of the claims as detailed, then Applicant is requested to amend the claims such that the invention is clearly stated in contrast to the admitted prior art of record, wherein the user performs the implementation. In order to limit the prosecution, the examiner has considered the claims with this distinction claimed. However, the examiner would like to make Applicant aware that the Admitted Prior Art essentially meets the claims, since the implementing steps can be performed in any manner in combination with the receiving step in order to achieve dynamic implementation of a Java Metadata Interface.

The examiner notes that Applicant's invention regarding claims 34 and 50 intends for **a determination of whether a JMI interface is implemented or unimplemented such that if the JMI interface is unimplemented, a JMI interface is implemented to be used and if the JMI interface is implemented, executing the stored JMI interface implementation without implementing a new JMI interface.** The claims as written allow for implementing a JMI interface if one is unimplemented and after the implementation is created and stored, executing calls to it and therefore has been examined as such. In addition, the examiner has understood the implementing to be performed automatically at run-time as detailed in the first observation. If Applicant agrees with the Examiner's assessment of the claims as detailed, then Applicant is requested to amend the claims such that the invention is clearly stated in contrast to the prior art of record.

### ***Respons to Arguments***

Art Unit: 2127

5. Applicant's arguments with respect to claims 1, 2, 5, 9, 17-19, 22, 26, 34, 35, 38, 42, 50-52, 56 and 59 have been considered but are moot in view of the new ground(s) of rejection.


***Allowable Subject Matter***

6. Claims 3, 4, 6-8, 10-16, 20, 21, 23-25, 27-33, 36, 37, 39-41, 43-49, 53, 54, 56-58 and 60-66 objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims and when the terminal disclaimer has been filed regarding the provisional double patenting.

***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Diem Cao whose telephone number is (571) 272-3760. The examiner can normally be reached on Monday-Friday, 8:30 am - 5:00 pm.

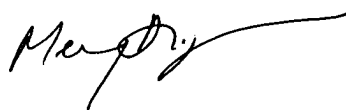
If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

  
MENG-AL T. AN  
SENIOR PATENT EXAMINER  
LOGY CENTER 2100

Art Unit: 2127

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

L. B.  
January 12, 2005

A handwritten signature in black ink, appearing to be "M. P. J.", with a long horizontal flourish extending to the right.